



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/015,958	10/30/2001	David George De Vorchik	MFCP.88142	6989
45809	7590	08/19/2005	EXAMINER	
SHOOK, HARDY & BACON L.L.P. 2555 GRAND BOULEVARD KANSAS CITY, MO 64108-2613			KISS, ERIC B	
			ART UNIT	PAPER NUMBER
			2192	

DATE MAILED: 08/19/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/015,958	DE VORCHIK ET AL.
	Examiner	Art Unit
	Eric B. Kiss	2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 16 May 2005.
- 2a) This action is FINAL. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-15 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-15 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____. | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. The reply filed 16 May 2005 has been received and entered. Claims 1-15 are pending.

Response to Amendment

2. Applicant's amendments to claim 4 appropriately address the objection to claim 4, based on an informality. Accordingly, this objection is withdrawn in view of Applicant's amendments.
3. Applicant's amendments to claims 1-6, 8, and 9 appropriately address the rejection of claims 1-6, 8, and 9 under 35 U.S.C. §112, second paragraph, based on indefiniteness. Accordingly, this rejection is withdrawn in view of Applicant's amendments.

Response to Arguments

4. Applicant's arguments filed 16 May 2005 have been fully considered but they are not persuasive.
 - a. In response to Applicant's arguments regarding claims 7, 10, and 11, the Examiner maintains that Fedorov discloses providing a host wizard; providing a web component comprising: a web page, said web page containing a wizard control area and a control interface. Fedorov discloses *seismic.asp*, an Active Server Page that causes the loading and display of a wizard dialog box (see, for example, the last paragraph on p. 423). It is noted that Applicant has not provided a definition of the term "host wizard" that excludes this interpretation of Fedorov.

Fedorov further illustrates, in the screenshots on pp. 424-426 the wizard dialog web page containing a wizard control area and a control interface, *e.g.*, the illustrated wizard has buttons that allow a user to control its actions.

Further, the Examiner maintains that Fedorov discloses the integration of the host wizard and the web component. For example, the user-interface is provided by *equakeget.htm*, an HTML page, which interacts with the user and submits entered results to the Active Server Page script in *seismic.asp* (see, for example, “Submitting the Data” on p. 431).

b. In response to Applicant’s arguments regarding claim 1, the Examiner maintains that Gauthier discloses invoking said one or more sub-wizard components during said host-wizard component execution; and transferring control from said host-wizard to said one or more sub-wizard components. As is disclosed in the cited section of Gauthier, the subwizards are separately launchable components within the overall wizard. Further, as disclosed in col. 14, line 9, through col. 15, line 24, the subwizard includes several objects that define its functionality. For example, the WizardState object included in the subwizard maintains a set of attributes used to get and channel data from default attributes and/or user input and deliver those attributes to a WizardCodeGenerator object (col. 14, lines 29-37), and the WizardCodeGenerator object uses this channeled data to either generate code (for example, in the context of a wizard designed to configure a peripheral device) or perform or execute existing code (col. 15, lines 1-24). As the functionality for the subwizard is defined within the subwizard, control must be passed to the subwizard during execution in order to realize the described functionality.

Contrary to Applicant's contention, claim 1 does not require a host wizard to transfer control *directly* to a sub-wizard, and the sub-wizard *controlling its own execution*.

During patent examination, the claims are given the broadest reasonable interpretation consistent with the specification. See *In re Morris*, 127 F.3d 1048, 44 USPQ2d 1023 (Fed. Cir. 1997).

c. In response to Applicant's arguments regarding claim 2, the Examiner maintains that XML is a "browser based" technology. The current XML specification ("Extensible Markup Language (XML) 1.0 (Third Edition): W3C Recommendation 04 February 2004," available on the web site of the World Wide Web Consortium at <URL: <http://www.w3.org/TR/2004/REC-xml-20040204/>> describes XML in this manner:

The Extensible Markup Language (XML) is a subset of SGML that is completely described in this document. Its goal is to enable generic SGML to be served, received, and processed on the Web in the way that is now possible with HTML. XML has been designed for ease of implementation and for interoperability with both SGML and HTML.

Further, it is noted that the term "browser based object component" is found within the original specification only in the context of original claims 2 and 5. No definition of "browser based object component" that excludes XML is found in the specification (including original claims 2 and 5), nor has Applicant advanced such an exclusive definition.

d. In response to Applicant's arguments regarding claim 3, the Examiner maintains that Gauthier discloses one or more subwizard components being operating system-based application component object extensions.

As the wizard components described in Gauthier are designed to execute on top of an operating system and provide additional functionality (e.g., enabling the display and interaction with user interface panels designed to facilitate code generation or execution), these components can be interpreted as operating system based extensions.

Further, it is noted that Applicant has not provided a definition of the term “operating system based” that excludes this interpretation of Gauthier.

e. In response to Applicant’s arguments regarding claim 4, as disclosed by Gauthier, the WizardManager class defines objects which control the execution of multiple subwizards within the target wizards. Gauthier further discloses a WizardManagerSelectionPanel class providing a GUI interface panel with dynamically updated content to allow a user of the target wizard to select from a list of available subwizards (col. 10, lines 30-33), thus providing the necessary navigation component to invoke the selected subwizard. Further, as disclosed in col. 14, line 9, through col. 15, line 24, the subwizard includes several objects that define its functionality. For example, the WizardState object included in the subwizard maintains a set of attributes used to get and channel data from default attributes and/or user input and deliver those attributes to a WizardCodeGenerator object (col. 14, lines 29-37), and the WizardCodeGenerator object uses this channeled data to either generate code (for example, in the context of a wizard designed to configure a peripheral device) or perform or execute existing code (col. 15, lines 1-24). As the functionality for the subwizard is defined within the subwizard, control must be passed to the subwizard during execution in order to realize the described functionality.

Art Unit: 2192

f. In response to Applicant's arguments regarding claim 12, the Examiner respectfully submits that the wizard components of Gauthier have defined panel flows that define how each panel is connected to a previous or next panel (as discussed in the cited sections).

Further, although not relied upon in the applied rejections, it is noted that such panel navigation is a standard feature of wizards and is admitted prior art discussed in the instant application's background section (p. 2, lines 13-17).

Claim Rejections - 35 USC § 102

5. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

6. Claims 7, 10, and 11 are rejected under 35 U.S.C. 102(b) as being anticipated by Alex Fedorov, et al., "Professional Active Server Pages 2.0," 1998, Wrox Press Ltd (hereinafter *Fedorov et al.*).

As per claim 7, *Fedorov et al.* discloses providing a host wizard (*seismic.asp*, an Active Server Page that causes the loading and display of a wizard dialog box; see, for example, the last paragraph on p. 423); providing a web component comprising: a web page containing a header area, a wizard control area and a control interface area (see, for example, the screenshots of the wizard dialog on pp. 424-426), the control interface area having navigation control adapted to recursively navigate within said web component and to said host wizard, by utilizing one or more object module functions enabling navigation (the screenshots of the wizard dialog on pp. 424-426 clearly show the "< Back" and "Next >" navigation controls). *Fedorov et al.* further

discloses providing a user interface that integrates the web component into the host wizard (the user-interface is provided by *equakeget.htm*, an HTML page, which interacts with the user and submits entered results to the Active Server Page script in *seismic.asp*; see, for example, “Submitting the Data” on p. 431); and providing an information container to exchange informational items between the web component and the host wizard (a return string is generated to convey the results; see, for example, “Submitting the Data” on p. 431).

As per claim 10, this is a computer readable medium version of the method discussed above (claim 7). The use of such a computer readable medium, such as memory, is further inherent in realizing the computer-implemented functionality disclosed by *Fedorov et al.*

As per claim 11, this is a computer system version of the method discussed above (claim 7). *Fedorov et al.* further discloses the prescribed methods as being computer-implemented (for example, the screenshots on pp. 424-427 illustrate execution within an Internet Explorer web browser environment, which inherently requires a processor and a memory to function as illustrated/described).

7. Claims 1-6, 8, 9, and 12-15 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent No. 6,574,791 to Gauthier et al.

As per claim 1, *Gauthier et al.* discloses providing a host-wizard component (see, for example, col. 9, line 56, through col. 10, line 5); providing one or more sub-wizard components (see, for example, col. 9, line 56, through col. 10, line 5); and said host-wizard invoking said one or more subwizard components during said host-wizard component execution (see, for example, col. 9, line 56, through col. 10, line 5); and transferring control from said host-wizard to said one

Art Unit: 2192

or more sub-wizard components. As disclosed in col. 14, line 9, through col. 15, line 24, the subwizard includes several objects that define its functionality. For example, the WizardState object included in the subwizard maintains a set of attributes used to get and channel data from default attributes and/or user input and deliver those attributes to a WizardCodeGenerator object (col. 14, lines 29-37), and the WizardCodeGenerator object uses this channeled data to either generate code (for example, in the context of a wizard designed to configure a peripheral device) or perform or execute existing code (col. 15, lines 1-24). As the functionality for the subwizard is defined within the subwizard, control must be passed to the subwizard during execution in order to realize the described functionality.

As per claim 2, *Gauthier et al.* further discloses the sub-wizard components being browser based object components (see, for example, col. 18, line 66, through col. 19, line 30).

As per claim 3, *Gauthier et al.* further discloses the sub-wizard components being operating system based application component object extensions (the basic functionality of the computer system disclosed by *Gauthier et al.* is controlled by operating system 100; see, for example, col. 6, lines 42-46).

As per claim 8, this is a computer readable medium version of the method discussed above (claim 1). *Gauthier et al.* further discloses the use of such a medium to implement the prescribed methods (see, for example, col. 5, line 52, through col. 6, line 4).

As per claim 9, this is a computer system version of the method discussed above (claim 1). *Gauthier et al.* further discloses the use of such a system to implement the prescribed methods (see, for example, col. 5, line 24, through col. 6, line 46).

As per claim 4, *Gauthier et al.* discloses a host wizard having a host-wizard interface adapted to communicate with other wizards (see, for example, col. 9, line 56, through col. 10, line 5) and a host-wizard navigational component adapted to transfer control to other wizards (see, for example, col. 10, lines 57-67; and col. 14, lines 9-14); one or more sub-wizard components, said one or more sub-wizard components having sub-wizard interfaces adapted to communication with other wizards and sub-wizard navigational components adapted to transfer control to other wizards. The WizardManager class disclosed by *Gauthier et al.* defines objects which control the execution of multiple subwizards within the target wizards. Gauthier further discloses a WizardManagerSelectionPanel class providing a GUI interface panel with dynamically updated content to allow a user of the target wizard to select from a list of available subwizards (col. 10, lines 30-33), thus providing the necessary navigation component to invoke the selected subwizard. Further, as disclosed in col. 14, line 9, through col. 15, line 24, the subwizard includes several objects that define its functionality. For example, the WizardState object included in the subwizard maintains a set of attributes used to get and channel data from default attributes and/or user input and deliver those attributes to a WizardCodeGenerator object (col. 14, lines 29-37), and the WizardCodeGenerator object uses this channeled data to either generate code (for example, in the context of a wizard designed to configure a peripheral device) or perform or execute existing code (col. 15, lines 1-24). As the functionality for the subwizard is defined within the subwizard, control must be passed to the subwizard during execution in order to realize the described functionality. *Gauthier et al.* further discloses said host wizard can communicate with one or more sub-wizard components through the host-wizard interface and sub-wizard interfaces (see, for example, col. 9, line 56, through col. 10, line 5; col. 10, lines 57-

67; and col. 14, lines 9-14); and wherein control is transferred between the host wizard and the sub-wizard component through the host navigational component and the sub-wizard navigational component (see, for example, col. 10, lines 57-67; and col. 14, lines 9-14).

As per claim 5, *Gauthier et al.* further discloses the sub-wizard components being browser based object components (see, for example, col. 18, line 66, through col. 19, line 30).

As per claim 6, *Gauthier et al.* further discloses the sub-wizard components being operating system based application component object extensions (the basic functionality of the computer system disclosed by *Gauthier et al.* is controlled by operating system 100; see, for example, col. 6, lines 42-46).

As per claim 12, *Gauthier et al.* discloses providing a first wizard (see, for example, col. 9, line 56, through col. 10, line 5); providing a second wizard (see, for example, col. 9, line 56, through col. 10, line 5); and providing at least one navigational component on each of said first and second wizards allowing sequential progression or regression through said first and second wizards to chain said second wizard to said first wizard (see, for example, col. 10, lines 57-67; and col. 14, lines 9-14).

As per claim 13, *Gauthier et al.* further discloses the first and second wizards each being an operating system based wizard (the basic functionality of the computer system disclosed by *Gauthier et al.* is controlled by operating system 100; see, for example, col. 6, lines 42-46) or a web based wizard (see, for example, col. 18, line 66, through col. 19, line 30).

As per claim 14, this is a computer system version of the method discussed above (claim 12). *Gauthier et al.* further discloses the use of such a system to implement the prescribed methods (see, for example, col. 5, line 24, through col. 6, line 46).

Art Unit: 2192

As per claim 15; this is a computer readable medium version of the method discussed above (claim 12). *Gauthier et al.* further discloses the use of such a medium to implement the prescribed methods (see, for example, col. 5, line 52, through col. 6, line 4).

Conclusion

8. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

9. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Eric B. Kiss whose telephone number is (571) 272-3699. The Examiner can normally be reached on Tue. - Fri., 7:00 am - 4:30 pm. The Examiner can also be reached on alternate Mondays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Tuan Dam, can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Any inquiry of a general nature should be directed to the TC 2100 Group receptionist:
571-272-2100.

EBK /EBK
August 12, 2005


TUAN DAM
SUPERVISORY PATENT EXAMINER